THIS OPINION WAS NOT WRITTEN FOR PUBLICATION

The opinion in support of the decision being entered today (1) was not written for publication in a law journal and (2) is not binding precedent of the Board.

Paper No. 16

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte AHARON AHARON, YOSSI MALKA, and YOSSI LICHTENSTEIN

Appeal No. 96-3762
Application No. 08/245,179[1]

ON BRIEF

Before BARRETT, FLEMING, and GROSS, Administrative Patent Judges.

GROSS, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on appeal from the examiner's final rejection of claims 1 through 15, which are all of the claims pending in this application.

[1] Application for patent filed May 17, 1994.

The appellants' invention relates to a generator of test programs for checking the operation of a hardware processor design. Claim 1 is illustrative of the claimed invention, and it reads as follows:

1. A test program generator for producing test programs for checking the operation of a hardware processor design, said test program generator comprising:

storing means for storing data representing a processor instruction set and resources, said stored data being represented as a separate declarative specification, said separate declarative specification being a representation of relationships between semantic entities associated with each instruction and between said semantic entities and said processor resources;

extracting means for extracting said data from said storage means and for transforming said data into internal data structures; and

test program generating means for generating test programs from said internal data structures.

The prior art references of record relied upon by the examiner in rejecting the appealed claims are:

Highland	4,924,408	May 08, 1990
Loopik et al. (Loopik)	5,381,417	Jan. 10, 1995
		(filed Aug. 19, 1988)

Claims 1 through 15 stand rejected under 35 U.S.C. § 103 as being unpatentable over Loopik in view of Highland.

Reference is made to the examiner's answer (Paper No. 11, mailed May 29, 1996) for the examiner's complete reasoning in support of the rejections, and to the appellants' brief (Paper No. 10, filed February 26, 1996) for the appellants' arguments thereagainst.

OPINION

We have carefully considered the claims, the applied prior art references, and the respective positions articulated by the appellants and the examiner. As a consequence of our review, we will reverse the obviousness rejection of claims 1 through 15.

The claims are drawn to a test program generator and method for creating test programs "for checking the operation of a hardware processor design." A processor, as broadly defined, is merely a device which processes input data. Loopik, the primary reference, discloses a test program generator for circuit assemblies, where "circuit assembly" refers to "a circuit board and also . . . smaller assemblies of components, such as multi-chip modules, designed to be mounted on a circuit board" (Loopik, column 1, lines 5-10). Since circuit boards process input data, "processor," as broadly defined, encompasses a circuit assembly, in the absence of further limitations which require more than merely processing input.

Figure 5 of Loopik shows a data base 62 for providing data to a test program generator 64, which meets the claimed "storing means." Although Loopik does not state that the data base stores data which is represented as a separate declarative specifi-ation, the definition given by appellants (specification, page 5) for "separate declarative specification" simply requires a data base. Accordingly, the data of Loopik's data base appears to be represented as a separate declarative specification.

Each independent claim recites "data representing a processor instruction set and resources" and "the declarative specification being a representation of relationships between semantic entities associated with each instruction and between said semantic entities and said processor resources." As stated above, the circuit assembly can be considered a processor, but only if no further claim limitations require more than merely processing data. The second paragraph, as quoted above, further distinguishes the processor. Accordingly, whether the circuit assembly of Loopik can serve as the claimed processor depends on whether the data of Loopik represents a processor instruction set and resources, and whether the data base is a representation between semantic entities and the processor resources.

The examiner has failed to indicate what elements of Loopik would satisfy the limitation of an instruction set and resources. Although the examiner attempts to combine Loopik with Highland, if Loopik does not include an instruction set and resources, Loopik does not disclose a processor, and therefore does not relate to a computer. Accordingly, it is unclear how one can modify the device with the computer of Highland.

Further, the examiner admits (Paper No. 5, page 3) that Loopik does not disclose or suggest "the data being represented by relationships between semantic entities associated with each instruction and between semantic entities and processor resources." The examiner relies on Highland to make up for this deficiency. Highland teaches converting the knowledge base of a computer system into program code in the form of rule trees. The examiner concludes that modifying the device of Loopik with the teachings of Highland yields the claimed representation of the data in the storing means. However, Loopik's knowledge base is not part of the data base (see Figure 5). Accordingly, implementing the teachings of Highland results in the knowledge base, but not the data in the data base, being represented by relationships between semantic entities associated with each instruction and between semantic entities and processor resources. Further, the examiner has not provided any other explanation as to why it would have been obvious to make the data in the data base be represented by the claimed relationships. Thus, the examiner has not met all of the elements of the claims, and consequently has failed to establish a prima facie case of obviousness. Therefore, the rejection cannot be sustained.

CONCLUSION

The decision of the examiner to reject claims 1 through 15 under 35 U.S.C. § 103 is reversed.

REVERSED

LEE E. BARRETT Administrative Patent Judge	)	
MICHAEL R. FLEMING Administrative Patent Judge	)	BOARD OF PATENT APPEALS AND INTERFERENCES
ANITA PELLMAN GROSS Administrative Patent Judge	)	

clm

Steven W. Roth
IBM Corp., Dept. 917
3605 Highway 52 N.
Rochester, MN 55901